**Installation of VS Code:**

**Steps to Download and Install Visual Studio Code on Windows 11:**

1. **Prerequisites:**
   * Windows 11 operating system.
   * Internet connection.
2. **Download Visual Studio Code:**
   * Visit the [Visual Studio Code download page](https://code.visualstudio.com/Download).
   * Click on the "Windows" download button to download the installer.
3. **Install Visual Studio Code:**
   * Once the installer is downloaded, run the installer file (VSCodeSetup.exe).
   * Follow the installation wizard steps:
     + Accept the license agreement.
     + Choose the destination folder.
     + Select additional tasks (e.g., create a desktop icon, add to PATH, etc.).
   * Click "Install" and wait for the installation to complete.
   * After installation, you can launch Visual Studio Code by clicking "Finish" or by finding it in the Start menu.

**First-time Setup:**

**Initial Configurations and Settings for an Optimal Coding Environment:**

1. **Select a Theme:**
   * Open the Command Palette (Ctrl+Shift+P).
   * Type "Preferences: Color Theme" and select it.
   * Choose a theme that you prefer (e.g., Dark+, Light+).
2. **Install Extensions:**
   * Python: Install the Python extension for Python development.
   * Prettier: Install Prettier for code formatting.
   * GitLens: Enhance Git capabilities within VS Code.
3. **Configure Settings:**
   * Open Settings (Ctrl+,).
   * Adjust font size, tab size, and other preferences under the Text Editor section.
   * Enable format on save under Editor: Format On Save.

**User Interface Overview:**

**Main Components of the VS Code User Interface:**

1. **Activity Bar:**
   * Located on the far left.
   * Contains icons for various views (Explorer, Search, Source Control, Run and Debug, Extensions).
   * Provides quick access to these views.
2. **Side Bar:**
   * Located next to the Activity Bar.
   * Displays the contents of the selected view from the Activity Bar (e.g., file explorer, search results).
3. **Editor Group:**
   * Main area where files are opened and edited.
   * Supports multiple editors in tabs and split view.
4. **Status Bar:**
   * Located at the bottom.
   * Displays information about the current file (e.g., line number, column number, programming language mode).
   * Shows notifications and actions (e.g., Git branch, errors).

**Command Palette:**

**What is the Command Palette and How to Access It:**

* **Command Palette:**
  + A powerful tool to access all available commands in VS Code.
  + Allows users to run commands, navigate to files, and change settings quickly.
* **Accessing the Command Palette:**
  + Press Ctrl+Shift+P or F1.
* **Examples of Common Tasks:**
  + Open a file: Type Open File and select the file.
  + Change settings: Type Preferences: Open Settings and modify settings.
  + Install extensions: Type Extensions: Install Extensions and search for the desired extension.

**Extensions in VS Code:**

**Role of Extensions and How to Manage Them:**

* **Role of Extensions:**
  + Enhance the functionality of VS Code by adding features for different languages, tools, and frameworks.
* **Finding and Installing Extensions:**
  + Open the Extensions view by clicking the Extensions icon in the Activity Bar or pressing Ctrl+Shift+X.
  + Search for extensions by name or keyword.
  + Click "Install" to add the extension to your setup.
* **Managing Extensions:**
  + Disable or uninstall extensions from the Extensions view.
  + Configure extension settings via the gear icon next to the extension name.
* **Examples of Essential Extensions for Web Development:**
  + **HTML:** HTML Snippets
  + **CSS:** IntelliSense for CSS
  + **JavaScript:** ESLint
  + **React:** React Native Tools

**Integrated Terminal:**

**How to Open and Use the Integrated Terminal:**

* **Opening the Integrated Terminal:**
  + Press Ctrl+`` (backtick) or go to View > Terminal`.
* **Using the Integrated Terminal:**
  + Perform command-line operations directly within VS Code.
  + Run scripts, execute git commands, and manage project dependencies.
* **Advantages of Using the Integrated Terminal:**
  + Seamless workflow without switching to an external terminal.
  + Access to the same project directory and environment.

**File and Folder Management:**

**Creating, Opening, and Managing Files and Folders:**

* **Creating a File or Folder:**
  + Right-click in the Explorer view and select New File or New Folder.
  + Use Ctrl+N for a new file.
* **Opening a File or Folder:**
  + Click on the file in the Explorer view.
  + Use Ctrl+O to open a file and Ctrl+K Ctrl+O to open a folder.
* **Managing Files and Folders:**
  + Move, rename, or delete files and folders via right-click context menu.
  + Use Ctrl+P to quickly navigate between files.

**Settings and Preferences:**

**Customizing Settings in VS Code:**

* **Accessing Settings:**
  + Open Settings via Ctrl+, or go to File > Preferences > Settings.
* **Changing Theme:**
  + In Settings, search for Color Theme and select your preferred theme.
* **Adjusting Font Size:**
  + Search for Editor: Font Size in Settings and set the desired size.
* **Modifying Keybindings:**
  + Go to File > Preferences > Keyboard Shortcuts or press Ctrl+K Ctrl+S.
  + Change keybindings by clicking on the existing shortcut and pressing the new keys.

**Debugging in VS Code:**

**Setting Up and Starting Debugging:**

* **Setup:**
  + Open the file you want to debug.
  + Click the Run and Debug icon in the Activity Bar or press F5.
* **Starting Debugging:**
  + Set breakpoints by clicking in the gutter next to the line numbers.
  + Use the Debug toolbar to control the debugging session (step over, step into, continue, etc.).
* **Key Debugging Features:**
  + Breakpoints, watch variables, call stack, and debug console.

**Using Source Control:**

**Integrating Git with VS Code for Version Control:**

* **Initializing a Repository:**
  + Open the Source Control view by clicking the Source Control icon in the Activity Bar.
  + Click "Initialize Repository" and select the folder to initialize.
* **Making Commits:**
  + Stage changes by clicking the + icon next to the files.
  + Enter a commit message and click the checkmark icon to commit.
* **Pushing Changes to GitHub:**
  + Open the integrated terminal and add a remote repository:

bash

git remote add origin <https://github.com/yourusername/your-repo.git>

* + Push changes:

bash

git push -u origin main

Top of Form

Bottom of Form

|  |  |
| --- | --- |
|  |  |